State Machines Intro

Last updated 1/10/20

State Machine Intro

- These slides introduce the basics of State Machines

- Upon completion: You should be able to identify circuits that form a state machine and differentiate between the two types of state machines

State Machine Intro

- Synchronous Sequential Circuit

  - Synchronous
    - Outputs change on a clock edge
    - One common clock

  - Sequential
    - Involves memory
    - Register(s)

  - 2 generalized types
    - Finite State Machines
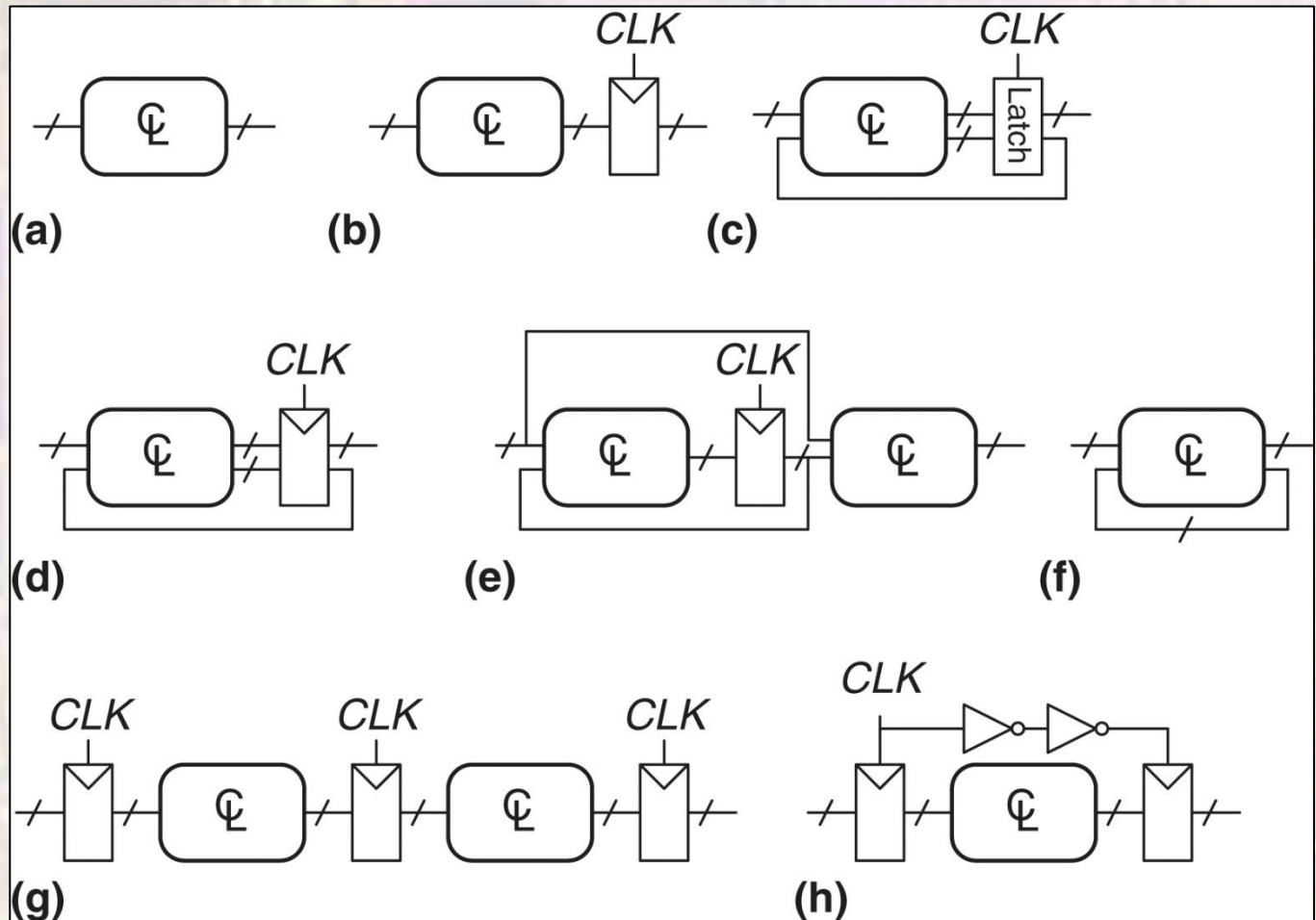   - Pipelined Machines

State Machine Intro

- Synchronous Sequential Circuit

  - Rules

    - Every element is either a
      - Register
      - Combinational Logic

    - At least one element is a register

    - All registers are driven by the same clock

    - Every cyclic path contains at least one register

State Machine Intro

- Synchronous Sequential Circuit

- Rules

src: Harris & Harris

© tj

State Machine Intro

- Synchronous Sequential Circuit

  - State
    - Collection of all unique values held in memory elements
    - 1 D-FF can hold 2 states (0 or 1)
    - 4 D-FFs can hold 16 states (0000, 0001, 0010, … 1111)

  - State Variable
    - Ordered collection of memory elements
    - Can hold any of 2^n states
    - 1 D-FF can have 1 state variable (Q)
    - 4 D-FFs can have 1 state variable (S) with 16 possible values

  Actually, there are many more possible values – what are they?

State Machine Intro

- ## Synchronous Sequential Circuit

  - ## State / Next State
    - State (S) - current value of a State Variable
    - Next State (S') - future value of a State Variable (next clock cycle)

    - If we know what S is, we can figure out what S' is
    - If we know what S' is, we know what S <u>WILL BE</u> after the next clock

S' S

inputs → **Combinational Logic** —n/→ **Register (D-FFs)** —n/→ outputs
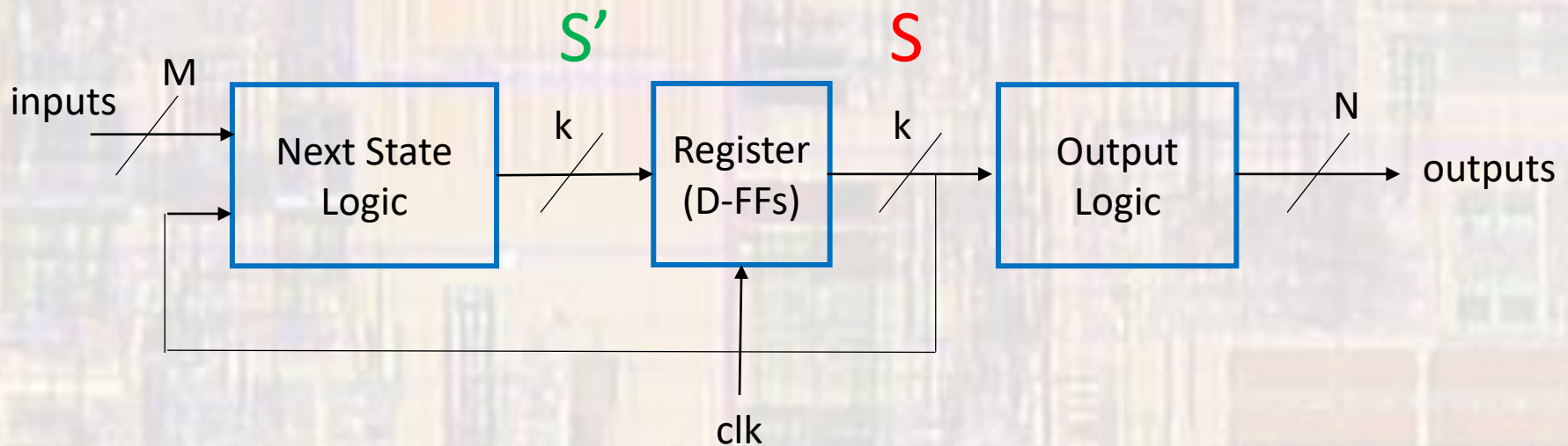
clk
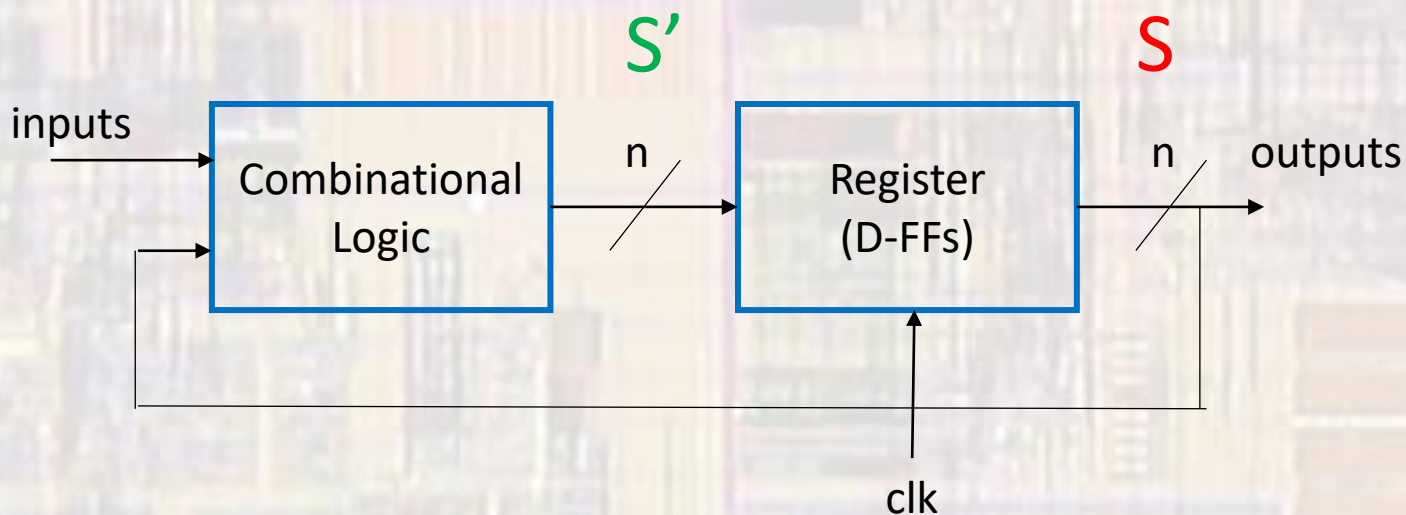
State Machine Intro

- Finite State Machine

  - Finite
    - n storage elements (D-FFs) $\rightarrow$ 2^n states – not infinite

  - State
    - Meets the requirements for a synchronous sequential circuit

  - 2 common types
    - Mealy Machine
    - Moore Machine

State Machine Intro

- Finite State Machine

  - Moore Machine

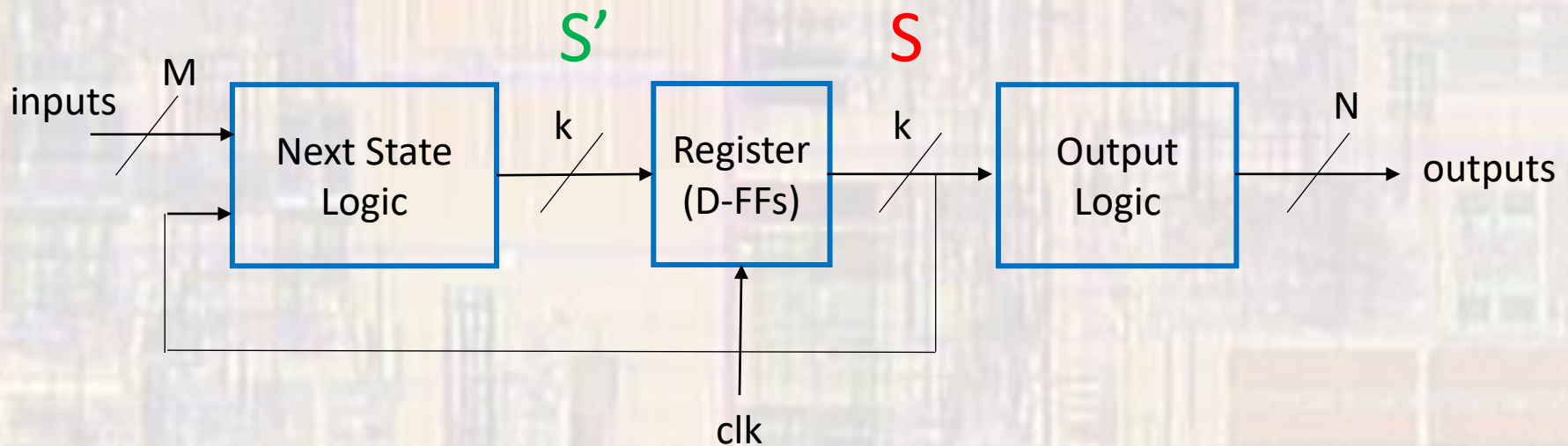   - Outputs depend only on the current state(S)

S' $\qquad$ S

inputs M → **Next State Logic** — k → **Register (D-FFs)** — k → **Output Logic** → N outputs

clk
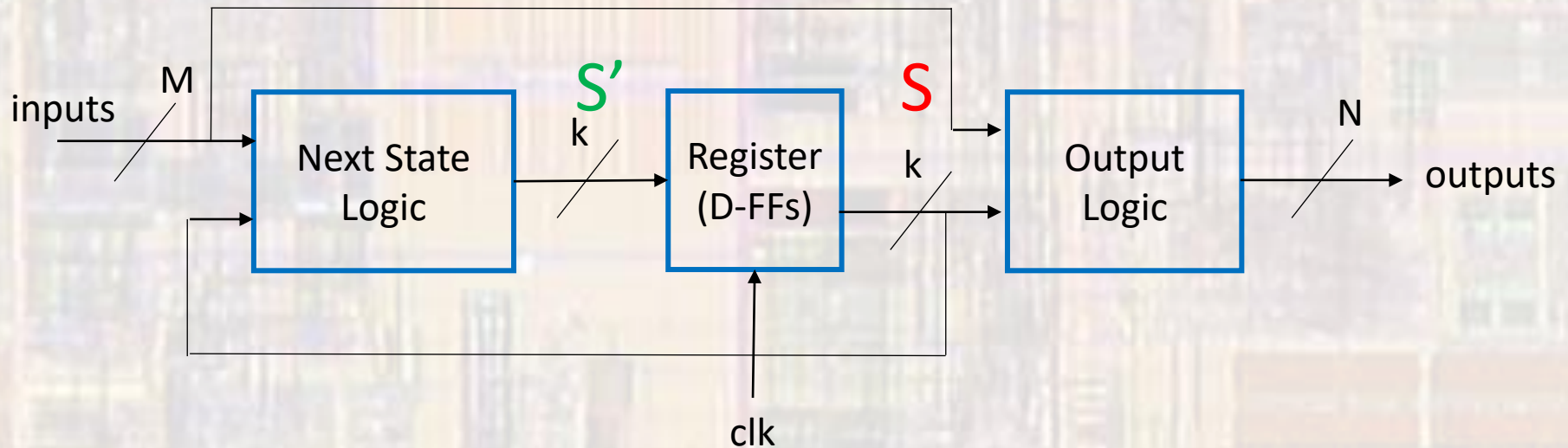
State Machine Intro

- Finite State Machine

  - Moore Machine

    - Changes in the input cause S' to change
    - When S' changes then the 'next state' S will be different than the current state S

S' S

inputs M

| Next State Logic | k | Register (D-FFs) | k | Output Logic | N | outputs |

clk

State Machine Intro

- ## Finite State Machine

  - ## Mealy Machine

    - Outputs depend on the current state(S) and the inputs

State Machine Intro

- Finite State Machine

  - Mealy Machine
    - Changes in the input cause S' to change
    - When S' changes then the 'next state' S will be different than the current state S